



Configuring CSS Network Protocols

This chapter describes how to configure the CSS DNS, ARP, RIP, IP, routing, bridging, SSH, and opportunistic Layer 3 forwarding functions. Information in this chapter applies to all CSS models except where noted.

This chapter includes the following sections:

- Configuring Domain Name Service
- Configuring Address Resolution Protocol
- Configuring Routing Information Protocol
- Configuring Internet Protocol
- Configuring an IP Route
- Configuring IP Source-Route
- Disabling an Implicit Service for Static Route Next Hop
- Configuring IP Subnet-Broadcast
- Showing IP Information
- Configuring Bridging for the CSS
- Configuring Secure Shell Daemon
- Configuring Opportunistic Layer 3 Forwarding

Configuring Domain Name Service

Use the **dns** command to enter commands that control Domain Name Service (DNS), the facility that translates host names such as myhost.mydomain.com to IP (Internet Protocol) addresses such as 192.168.11.1. The options for this global configuration mode command are:

- **dns primary** - Specify the primary DNS server
- **dns resolve** - Query DNS to resolve a hostname
- **dns secondary** - Specify the secondary DNS server
- **dns suffix** - Specify the default suffix to use when querying DNS
- **dnsflow** - Set up UDP traffic to DNS server port 53 as a CSS flow or forwards the traffic

Use the **show running-config global** command to display DNS configurations (refer to “Using the Running-Config and Startup-Config” in Chapter 1, Logging in and Getting Started).

Specifying a Primary DNS Server

To specify the primary DNS server, use the **dns primary** command followed by the IP address of the DNS server you wish to specify as the primary DNS server. Enter the IP address in dotted-decimal notation (for example, 192.168.11.1).

For example:

```
(config)# dns primary 192.168.11.1
```

To remove the primary DNS server, enter:

```
(config)# no dns primary
```

Using DNS Resolve

To resolve a hostname by querying the DNS server, use the **dns resolve** command followed by the host name you want to resolve. Enter the host name in mnemonic host-name format (for example, myhost.mydomain.com).

For example:

```
(config)# dns resolve fred.arrowpoint.com
```

Specifying a Secondary DNS Server

When a primary DNS server fails, the CSS uses the secondary DNS server to resolve host names to IP addresses. To specify a secondary DNS server, use the **dns secondary** command followed by the IP address of the secondary DNS server. Enter the IP address in dotted-decimal notation (for example, 192.168.11.1).

```
(config)# dns secondary 192.158.3.6
```



Note

You can specify a maximum of two secondary servers. To specify each additional server, repeat the **dns secondary** command. The order in which you enter the IP addresses is the order in which they are used.

To remove a secondary DNS server, enter the **no** version of the command followed by the IP address of the DNS server you wish to remove. For example:

```
(config)# no dns secondary 192.158.3.6
```

Specifying a DNS Suffix

To specify the default suffix to use when querying the DNS facility, use the **dns suffix** command followed by the suffix you wish to use. Enter the default suffix as an unquoted text string with no spaces and a maximum length of 64 characters.

For example:

```
(config)# dns suffix arrowpoint.com
```

To remove the default DNS suffix, enter:

```
(config)# no dns suffix
```

Specifying UDP Traffic on the DNS Server Port

For DNS UDP traffic on port 53, use the **dnsflow** command to determine whether the CSS uses flow control blocks (FCBs) for DNS requests and responses. This command provides the following options:

- **enable** (default) - Causes the CSS to set up flows using FCBs for DNS requests and responses. Because UDP traffic is connectionless, the DNS flows remain active until the flow manager reclaims the flow resources.
- **disable** - Causes the CSS to not use FCBs for the DNS requests and responses. Use this setting for sites with heavy DNS traffic or sites where the DNS clients use a source and destination port of 53.

For example:

```
(config)# dnsflow disable
```

Configuring Address Resolution Protocol

Use the **arp** command and its options to statically configure the IP to Media Access Control (MAC) translations necessary for the CSS to send data to network nodes. The following sections discuss configuring Address Resolution Protocol (ARP) for the CSS.

- Configuring ARP
- Configuring ARP Timeout
- Configuring ARP Wait
- Updating ARP Parameters
- Clearing ARP Parameters
- Showing ARP Information

Configuring ARP

To define a static ARP mapping, use the **arp** command. The syntax for this global configuration mode command is:

- **arp** *ip_address mac_address interface {vlan}*
- **arp** *hostname mac_address interface {vlan}*

The variables and options are:

- *ip_address* - The address of the system for static mapping. Enter an IP address in dotted-decimal notation (for example, 192.168.11.1) or in mnemonic host-name format (for example, myhost.mydomain.com).
- *hostname* - The address of the system for static mapping. Enter a hostname in mnemonic host-name format (for example, myhost.mydomain.com). You must configure DNS and the hostname must be resolved to an IP address for hostname to work.
- *interface* - The CSS interface that you want to configure. For a CSS 11050 or CSS 11150, enter the interface name in *interface port* format (for example, e2). For a CSS 11800, the interface format is *slot/port* (for example, 3/1).

- *vlan* - The number of the VLAN configured in a trunked interface on which this ARP address is configured (assuming trunking is enabled for the CSS Gigabit Interface port, see “Specifying VLAN Trunking to an Interface” in Chapter 4, Configuring Interfaces and Circuits). This argument is optional. Enter an integer from 1 to 4094 as the VLAN number.

To show static ARP mapping when you use the **show arp** command, the IP route must exist in the routing table.

For example:

```
(config)# arp 192.168.11.1 00-60-97-d5-26-ab ethernet-2
```

To remove a static mapping address, use the **no arp** command. For example:

```
(config)# no arp 192.168.11.1
```

The CSS discards ARP requests from hosts that are not on the same network as the CSS circuit IP address. Thus, if a CSS and a host are within the same VLAN but configured for different IP networks, the CSS does not respond to ARP requests from the host.

Configuring ARP Timeout

To set the time in seconds to hold an ARP resolution result, use the **arp timeout** command. When you change the timeout value, it only affects new ARP entries. All previous ARP entries retain the old timeout value. To remove all entries with the old timeout value, enter the **clear arp cache** command.

The timeout value is the number of seconds the CSS holds an ARP resolution result. To set a timeout period, enter an integer from 60 to 86400 (24 hours) seconds. The default is 14400 seconds (4 hours). If you do not want the ARP entries to timeout, enter **none** or **86401**.

For example:

```
(config)# arp timeout 120
```

To restore the default timeout value of 14400 seconds, enter:

```
(config)# no arp timeout
```

Configuring ARP Wait

To set the time in seconds to wait for an ARP resolution, use the **arp wait** command with a wait time. The wait time is the number of seconds the CSS waits for an ARP resolution in response to an ARP request to the network. Enter an integer from 5 to 30 seconds. The default is 5.

For example:

```
(config)# arp wait 15
```

To restore the default wait time of 5 seconds, enter:

```
(config)# no arp wait
```

Updating ARP Parameters

To update the file containing hosts reachable through ARP, use the **update arp** command. This command is available in SuperUser mode. For example:

```
# update arp file
```

Clearing ARP Parameters

The CSS enables you to clear ARP parameters for the ARP file or ARP cache. To clear the file that contains known hosts reachable through ARP, use the **clear arp file** command. For example:

```
clear arp file
```

To delete dynamic entries from the ARP cache, use the **clear arp cache** command with an IP address or hostname. The syntax and options for this command are:

- **clear arp cache** - Clear the entire ARP cache
- **clear arp cache *ip_address*** - Clear a single ARP IP address entry
- **clear arp cache *hostname*** - Clear a single ARP hostname entry

For example:

```
# clear arp cache
```

Showing ARP Information

To display ARP information, use the **show arp** command. The syntax and options for the command are:

- **show arp** - Display the complete ARP resolution table with IP addresses, MAC addresses, and resolution type.
- **show arp config** - Display ARP global configuration parameters. The screen displays the response timeout and the flush timeout in seconds.
- **show arp file** - Display the hosts reachable using ARP. The screen displays the IP addresses of the host systems.
- **show arp ip_address** - Display the resolution for the IP address.
- **show arp hostname** - Display the resolution for the hostname.

To display the complete ARP resolution table, enter:

```
# show arp
```

Table 3-1 describes the fields in the **show arp** output.

Table 3-1 Field Descriptions for the show arp Command

Field	Description
IP Address	The IP address of the system for static mapping.
MAC Address	The MAC address of the system mapped to the IP address.
Type	The resolution type for the entry. Dynamic indicates that the entry was discovered through the ARP protocol. Static indicates that the resolution is from a static configuration.
Port	The CSS interface configured as the egress logical port.

To display the global ARP configuration, enter:

```
# show arp config
```

Table 3-2 describes the fields in the **show arp config** output.

Table 3-2 *Field Descriptions for the show arp config Command*

Field	Description
ARP Response Timeout	The time in seconds to wait for an ARP resolution response before discarding the packet waiting to be forwarded to an address. The time can be from 5 to 30 seconds. The default is 5 seconds.
ARP Flush Timeout	The time in seconds to hold an ARP resolution result in the ARP cache. The timeout period can be from 60 to 86400 (24 hours). The default is 14400 (4 hours). An entry of none or 86401 indicates that the ARP entries will not timeout.

To display the host IP addresses entered at initialization or boot time through ARP, enter:

```
# show arp file
```

To display the resolution for a host IP address, enter:

```
# show arp 192.50.1.6
```

Configuring Routing Information Protocol

The CSS enables you to configure the following global Routing Information Protocol (RIP) attributes:

- **rip advertise** - Advertise a route through RIP on the CSS
- **rip redistribute** - Advertise routes from other protocols through RIP
- **rip equal-cost** - Specify how many equal-cost routes RIP can insert into the routing table

By default, RIP advertises RIP routes and local routes for interfaces running RIP. The **rip** command advertises other routes.

Configuring RIP Advertise

To advertise a route through RIP on the CSS, use the **rip advertise** command. The syntax for this command is:

```
rip advertise ip_address subnet_mask metric
```

- *ip_address* - The IP address for the route prefix. Enter an IP address in dotted-decimal notation (for example, 192.168.1.0).
- *subnet_mask* - The IP prefix length in CIDR bitcount notation (for example, /24) or in dot-decimal notation (for example, 255.255.255.0).
- *metric* - The optional metric to use when advertising this route. Enter a number from 1 to 15. The default is 1.

For example:

```
(config)# rip advertise 192.168.1.0/24 9
```



Note

The network does not have to be present in the routing table to be advertised. The **SNTPip advertise** command is intended for advertising Virtual IP addresses (VIPs).

To stop advertising a route through RIP on the CSS, enter:

```
(config)# no rip advertise 192.168.1.0/24
```

Configuring RIP Redistribute

To advertise routes from other protocols through RIP, use the **rip redistribute** command. By default, RIP advertises RIP routes and local routes for interfaces running RIP. This command instructs RIP to advertise other routes.

You can configure the following options for **rip redistribute**:

- **rip redistribute firewall *metric*** - Advertise firewall routes through RIP
- **rip redistribute local *metric*** - Advertise local routes (interfaces *not* running RIP)
- **rip redistribute static *metric*** - Advertise static routes
- **rip redistribute ospf *metric*** - Advertise OSPF routes

You can also enter an optional metric, which is the metric the CSS uses when advertising this route. Enter a number from 1 to 15. The default is 1.

For example:

```
(config)# rip redistribute static 3
```

To stop advertising routes from other protocols through RIP, use either the **local**, **static**, or **firewall** option.

The following command stops advertising static routes:

```
(config)# no rip redistribute firewall
(config)# no rip redistribute local
(config)# no rip redistribute static
(config)# no rip redistribute ospf
```

Configuring RIP Equal-Cost

To set the maximum number of routes RIP can insert into the routing table, use the **rip equal-cost** command. Enter a number from 1 to 15. The default is 1. For example:

```
(config)# rip equal-cost 4
```

To reset the number of routes to the default value of 1, enter:

```
(config)# no rip equal-cost
```

Showing RIP Configurations

To show a RIP configuration for one IP address or all IP addresses configured in the CSS, use the **show rip** command. This command provides the following options:

- **show rip** - Displays RIP configurations for all interfaces
- **show rip ip_address** - Displays a single RIP interface entry
- **show rip globals** - Displays RIP global statistics
- **show rip statistics** - Displays RIP interface statistics for all interfaces
- **show rip statistics ip_address** - Displays RIP interface statistics for a specific interface

Table 3-3 describes the fields in the **show rip** output.

Table 3-3 Field Descriptions for the show rip Command

Field	Description
IP Address	The advertised RIP interface address.
State	The operational state of the RIP interface.
RIP Send	The RIP version that the interface sends. The possible field values are: <ul style="list-style-type: none"> • none, do not send RIP packets • RIPv1, send RIP version 1 packets only • RIPv2, send RIP version 2 packets only (default)
RIP Recv	The RIP version that the interface receives. The possible values are: <ul style="list-style-type: none"> • both, receiving both version 1 and version 2 (default) • none, receiving no RIP packets • Ripv1, receiving RIP version 1 packets only • Ripv2, receiving RIP version 2 packets only
Default Metric	The default metric used when advertising the RIP interface.

Table 3-3 *Field Descriptions for the show rip Command (continued)*

Field	Description
Tx Log	The setting for the logging of RIP packet transmissions (enabled or disabled). The default setting is disabled.
Rx Log	The setting for the logging of RIP packet received (enabled or disabled). The default setting is disabled.

To display global RIP statistics, enter:

```
# show rip globals
```

Table 3-4 describes the fields in the **show rip globals** output.

Table 3-4 *Field Descriptions for the show rip globals Command*

Field	Description
RIP Route Changes	The global number of route changes made to the IP route database by RIP
RIP Query Responses	The global number of query responses sent to RIP query from other systems

To display the RIP interface statistics for all RIP interface entries, enter:

```
# show rip statistics
```

Table 3-5 describes the fields in the **show rip statistics** output.

Table 3-5 *Field Descriptions for the show rip statistics Command*

Field	Description
System Route Changes	The global number of route changes made to the IP route database by RIP
System Global Query Responses	The global number of query responses sent to RIP query from other systems
IP Address	The RIP interface IP address
Triggered Updates Sent	The number of triggered RIP updates sent by the interface

Table 3-5 Field Descriptions for the `show rip statistics` Command (continued)

Field	Description
Bad Packets Received	The number of bad RIP response packets received by the interface
Bad Routes Received	The number of bad routes in valid RIP packets received by the interface

Configuring Internet Protocol

To enter Internet Protocol (IP) configuration commands for the CSS, use the **ip** command. This command is available in configuration mode. The options for this command are:

- **ip record-route** - Enable processing of frames with a record-route option
- **ip redundancy** - Enable CSS-to-CSS redundancy
- **ip ecmp** - Set the equal-cost multipath selection algorithm

Configuring IP Record-Route

To enable the CSS to process frames with a record-route option, use the **ip record-route** command. For example:

```
(config)# ip record-route
```



Caution

Enabling **ip record-route** could pose security risks to your network. Record-route inserts the IP address of each router along a path into the IP header.

To disable processing frames with a record-route option (the default behavior), enter:

```
(config)# no ip record-route
```

Configuring IP Redundancy

To enable CSS-to-CSS redundancy, use the **ip redundancy** command. For example:

```
(config)# ip redundancy
```

To disable CSS-to-CSS redundancy, enter:

```
(config)# no ip redundancy
```

For information on configuring CSS-to-CSS redundancy, refer to the *Content Services Switch Advanced Configuration Guide*, Chapter 5, Configuring Redundant Content Services Switches.

Configuring IP ECMP

Use the **ip ecmp** command to set the equal-cost multipath selection algorithm and the preferred reverse egress path. The syntax and options for this global configuration mode command are:

- **ip ecmp address** - Choose among alternate paths based on IP addresses. For example:

```
(config)# ip ecmp address
```

- **ip ecmp no-prefer-ingress** - Do not prefer the ingress path of a flow for its reverse egress path. By default, the ingress path for a flow is its preferred egress path. For example:

```
(config)# ip ecmp no-prefer-ingress
```

To reset the ingress path of a flow for its preferred reverse egress path, enter:

```
(config)# no ip ecmp no-prefer-ingress
```

- **ip ecmp roundrobin** - Alternate between equal paths in roundrobin fashion. For example:

```
(config)# ip ecmp roundrobin
```

**Note**

The equal-cost multipath selection algorithm for non-TCP/UDP packets (for example, ICMP) is applied on a packet-by-packet basis. Multipath selection for TCP and UDP is performed on a per-flow basis and all packets for a particular flow take the same path.

ECMP cannot recover a failed router unless you configure a content rule for a router service.

Configuring an IP Route

A static route consists of a destination network address and mask, as well as the next hop to reach the destination. You can also specify a default static route (using 0.0.0.0 as the destination network address and a valid next hop address) to direct frames for which no other destination is listed in the routing table. Default static routes are useful for forwarding otherwise unrouteable packets by the CSS.

When you configure a static route, the CSS creates an internal service that periodically polls the configured next hop address with an ICMP echo (or ping) keepalive. The internal service is called an implicit service. If the router fails, the CSS removes any entries from the routing table that point to the failed router and stops sending network traffic to the failed router. When the router recovers, the CSS:

- Becomes aware of the router
- Reenters applicable routes into the routing table

The implicit service does not determine if the default or static route appears in the routing table. This decision is based on the CSS having a viable ARP entry for the next hop router IP address so the CSS can forward traffic to that destination. The CSS uses the ICMP keepalive as a means to ensure the next hop router MAC address is available and current. However, in certain situations, the next hop router may block ICMP message transmitted by the CSS, which results in a failed ICMP keepalive (the ICMP keepalive is in the Down state). As long as the CSS has the ARP entry of the next hop router the static route is still placed in the routing table.

**Note**

The CSS allows you to disable the internal ICMP keepalive through the **ip-no-implicit service** command. In this case, if the MAC address for the next hop is not known to the CSS the address will not appear in the routing table.

Use the **ip route** command to configure an IP route. You can configure a static route, a default static IP route, a blackhole route (where the CSS drops any packets addressed to the route), or a firewall IP route. Each **ip route** command requires either an:

- IP address and a subnet mask prefix - For example, 192.168.1.0/24
or
- IP address and a subnet mask - For example, 192.168.1.0 255.255.255.0

The **ip route** options are defined below. Note that the examples use the /subnet mask prefix option.

- **ip route** *IP address subnet mask blackhole* - Instructs the CSS to drop any packets addressed to the route. For example:

```
(config)# ip route 192.168.1.0/24 blackhole
```
- **ip route** *IP address subnet mask IP address2* - Specify the next hop address for the route. For example:

```
(config)# ip route 0.0.0.0/0 10.0.1.1
```
- **ip route** *IP address subnet mask IP address2 distance* - Specify the administrative distance. Enter an integer from 1 to 254. Note that the larger the administrative distance value (more hops), the less the route is preferred. For example:

```
(config)# ip route 0.0.0.0/0 10.0.1.1 40
```
- **ip route** *IP address subnet mask firewall index distance* - Configure a firewall route. The firewall option instructs the CSS to use firewall load balancing for this route. You can optionally set the administrative distance. For example:

```
(config)# ip route 192.168.1.0/24 firewall 3 2
```

- **ip route** *IP address subnet mask IP address* **originated-packets** - Specifies that the route is used only by packets that are created using flows or sessions going to and from the CSS (for example, a Telnet session to the CSS). The route is not used by flows or sessions that go through the CSS (for example, between an attached server and a remote client).

The optional **originated-packets** keyword instructs the CSS to use this route for flow and session packets going to and from the CSS (for example, a Telnet session to the CSS). Flows or session packets that go through the CSS (for example, between an attached server and a remote client) do not use this route. For example:

```
(config)# ip route 0.0.0.0/0 10.0.1.1 originated-packets
```

**Note**

Ping responses and SNMP responses do not use the originated-response route. Ping *requests* sent from the CSS use the originated-response route. Ping *responses* sent from the CSS do not use the originated-response route.

The variables are:

- *ip_address* - The destination network address. Enter the IP address in dotted-decimal notation (for example, 192.168.11.1).
- *subnet_mask* - The IP subnet mask. Enter the mask in either:
 - CIDR bitcount notation (for example, /24). Do not enter a space to separate the IP address from the prefix length.
 - Dotted-decimal notation (for example, 255.255.255.0).
- *ip_address2* - The next hop address for the route. Enter the IP address in dotted-decimal notation (for example, 192.168.11.1).
- *distance* - The optional administrative distance. Enter an integer from 1 to 254. A smaller number is preferable. The default value is 1.
- *index* - An existing index number for the firewall route. For information on configuring a firewall index, refer to the **ip firewall** command.

To remove a static route, enter:

```
(config)# no ip route 0.0.0.0/24 10.0.1.1
```

To disable the dropping of packets to a black-hole route, enter:

```
(config)# no ip route 192.168.1.0/24 blackhole
```

To remove a firewall route, enter:

```
(config)# no ip route 192.168.1.0/24 firewall 3
```

Configuring IP Source-Route

To enable processing of source-routed frames, use the **ip source-route** command. For example:

```
(config)# ip source-route
```



Caution

Enabling **ip source-route** could pose a major security risk to your network. Source-route specifies information that overrides the default routing a packet would normally take. The packet could then bypass a firewall.

To disable processing of source-routed frames (the default behavior), enter:

```
(config)# no ip source-route
```

Disabling an Implicit Service for Static Route Next Hop

Use the **ip no-implicit-service** command when you do not want the CSS to start an implicit service for the next hop of a static route. By default, the CSS establishes an implicit (or internal) service for the gateway address when a static route is defined. The **ip no-implicit-service** command specifies that no implicit service is established to the next hop of the static route, which disables the internal service ICMP keepalive. In this case, if the ARP address for the next hop is not known to the CSS the address will not appear in the routing table.

The purpose of the implicit service to the next hop of a static route is to monitor the availability of the next hop to forward data traffic. When the **ip no-implicit-service** command is in effect, traffic will be forwarded to the next hop even when the next hop is unavailable. Because of the possibility of data being lost if the next hop becomes unavailable, use of the **ip no-implicit-service** command is strongly discouraged.



Note

Static routes can sometimes appear in the CSS routing table even when you have an implicit service for the next hop address (the default setting) and the internal keepalive is down. When the CSS detects the ARP mapping for the next hop in the static route, the CSS continues to list that route in the routing table regardless of the state of the ICMP service keepalive (Down or Up).

When you implement the **ip no-implicit-service** global configuration command, this action does not affect previously configured static routes. The **ip no-implicit-service** command affects only those static routes added after you enable the command. Cisco Systems recommends you reboot the CSS after you modify the configuration to ensure all static routes are the same, which is useful for network monitoring and troubleshooting. If you wish to stop the implicit service for a previously configured static route, then you must delete and reconfigure the static route.

For example:

```
(config)# ip no-implicit-service
```

To reset the default setting (no implicit service is established to the next hop of the static route), enter:

```
(config)# no ip no-implicit-service
```

Configuring IP Subnet-Broadcast

To enable the CSS to forward subnet broadcast addressed frames, use the **ip subnet-broadcast** command.

For example:

```
(config)# ip subnet-broadcast
```

To disable forwarding of subnet broadcast addressed frames (the default behavior), enter:

```
(config)# no ip subnet-broadcast
```



Caution

Enabling the CSS to forward the subnet broadcast can make the subnet susceptible to “smurf” attacks; an attacker sends an ICMP echo request frame using a subnet broadcast address as a destination and a forged address as the source. If the attack is successful, all the destination subnet hosts reply to the echo and flood the path back to the source. By disabling the subnet broadcast forwarding, the original echo never reaches the hosts.

Showing IP Information

Use the **show ip** command to display Internet Protocol (IP) information for the CSS. Refer to the following sections to display CSS IP information.

- **Showing IP Config** - Display IP global configuration parameters
- **Showing IP Interfaces** - Display configured IP interfaces
- **Showing IP Routes** - Display IP routing information
- **Showing IP Statistics** - Display aggregate UDP and TCP statistics for the unit
- **Showing IP Summary** - Display a summary of IP global statistics

Showing IP Config

Use the **show ip config** command to display IP global configuration parameters. The parameters shows the state (enabled or disabled) of the source route option, forward IP broadcasts, record route option, and IP route change logging. It also shows the value for the orphaned route timer.

Table 3-6 describes the fields in the **show ip config** output.

Table 3-6 *Field Descriptions for the show ip config Command*

Field	Description
Source Route Option	Whether the processing of source-routed frames is enabled or disabled.
Forward IP Broadcasts	Whether the forwarding of IP broadcasts is enabled or disabled.
Orphaned Route Timer	The setting for the orphaned route timer.
Record Route Option	Whether the processing with a record-route option is enabled or disabled.

Table 3-6 Field Descriptions for the `show ip config` Command (continued)

Field	Description
Multiple Equal Cost Path Algorithm	The setting for the equal-cost multipath selection algorithm. The possible settings are: <ul style="list-style-type: none"> • Address, choose among alternate paths based on IP addresses • roundrobin, alternate between equal paths in roundrobin fashion
IP Route Change Logging	Whether the logging of IP route changes is enabled or disabled.

Showing IP Interfaces

Use the **show ip interfaces** command to display configured IP interfaces on the CSS. The display includes the circuit state, IP address, broadcast address, Internet Control Message Protocol (ICMP) settings, and Router Discovery Program (RDP) settings.

Table 3-7 describes the fields in the **show ip interfaces** output.

Table 3-7 Field Descriptions for the `show ip interfaces` Command

Field	Description
Circuit Name	The name of the circuit associated with the IP interface.
State	The state of the IP interface. The possible states are: <ul style="list-style-type: none"> • active (1), the interface is up • disabled (2), the interface is disabled • noCircuit (3), the interface is waiting for an underlying circuit
IP Address	The IP address assigned to the circuit.
Network Mask	The network mask of the circuit.

Table 3-7 Field Descriptions for the show ip interfaces Command (continued)

Field	Description
Broadcast Address	The broadcast IP address associated with the IP interface. If left at zero, the all-ones host is used for numbered interfaces. 255.255.255.255 is always used for unnumbered interfaces.
Redundancy	Indicates whether the redundancy protocol is running on the interface. The default state is disable.
ICMP Redirect	Whether the transmission of Internet Control Message Protocol (ICMP) redirect messages is enabled or disabled. The default state is Enabled.
ICMP Unreachable	Whether the transmission of ICMP “destination unreachable” messages is enabled or disabled. The default state is Enabled.
RIP	Whether the RIP is enabled or disabled.

Showing IP Routes

Use the **show ip routes** command to display IP routing information. The syntax and options for this command are:

- **show ip routes** - Display the entire routing table, including host IP address, next hop, interface, route type, protocol, age (in seconds), and metric
- **show ip routes firewall** - Display all firewall routes
- **show ip routes local** - Display all local routes
- **show ip routes ospf** - Display all OSPF routes
- **show ip routes rip** - Display all RIP routes
- **show ip routes static** - Display all static routes
- **show ip routes ip_address** or **host** {to *ip_address* or *host|mask* or *prefix*} - Display information about a route to a destination, a specific route, or routes in a range

The variables are:

- *ip_address* or *host* - The IP address of the host or network prefix. Enter an IP address in dotted-decimal notation (for example, 192.168.11.1). The IP address after the keyword is the last IP address in a range.
- *mask* or *prefix* - Subnet address of the specific network. Enter the subnet address in mask or prefix notation (for example, /24).

For example, to show all IP routes in the CSS, enter:

```
# show ip routes

Prefix/Length  Next Hop           if  Type    Proto  Age  Metric
172.16.0.0/16  172.16.59.12/16   14  mgmt    local
0.0.0.0/0      192.168.1.206     15  remote  rip    5    2
5.0.0.0/8      192.168.1.205     15  remote  rip    3    3
6.0.0.0/8      192.168.1.205     15  remote  rip    3    3
10.0.0.0/8     192.168.1.205     15  remote  rip    3    2
11.0.0.0/8     11.0.3.204        16  local   local  840  0
20.0.0.0/8     192.168.1.205     15  remote  rip    3    2
```

Table 3-8 describes the fields in the **show ip routes** output.

Table 3-8 Field Descriptions for the show ip routes Command

Field	Description
prefix/length	The IP address and prefix length for the route.
next hop	The IP address for the next hop.
if	The ifIndex value that identifies the local interface through which the next hop of this route should be reached.
type	The type of the route entry. The possible types are: <ul style="list-style-type: none"> • local, local interface • remote, remote destination • mgmt, management interface
proto	The protocol for the route.
age	The maximum age for the route.
metric	The metric cost for the route.

Showing IP Statistics

Use the **show ip statistics** command to display aggregate TCP statistics for the unit. Table 3-9 describes the fields in the **show ip statistics** output.

Table 3-9 *Field Descriptions for the show ip statistics Command*

Field	Description
UDP Statistics:	
Input Datagrams:	The total number of UDP datagrams delivered to UDP users.
No Port Errors:	The total number of received UDP datagrams for which there was no application at the destination port.
Output Datagrams:	The total number of UDP datagrams sent from the CSS.
Input Errors:	The number of received UDP datagrams that could not be delivered for reasons other than the lack of an application at the destination port.
TCP Statistics:	
Retransmit Algorithm:	The algorithm used to determine the timeout value for retransmitting unacknowledged octets.
Max Retransmit Time:	The maximum value permitted by a TCP implementation for the retransmission timeout, measured in milliseconds.
Active Opens:	The number of times TCP connections have made a direct transition to the SYN-SENT state from the CLOSED state.
Failed Attempts:	The number of times TCP connections have made a direct transition to the CLOSED state from either the SYN-SENT state or the SYN-RCVD state, plus the number of times TCP connections have made a direct transition to the LISTEN state from the SYN-RCVD state.
Established Conns:	The number of TCP connections for which the current state is either ESTABLISHED or CLOSE-WAIT.

Table 3-9 Field Descriptions for the show ip statistics Command (continued)

Field	Description
Output Segments:	The total number of segments sent, including those on current connections but excluding those containing only retransmitted octets.
Input Errors:	The total number of segments received in error (for example, bad TCP checksums).
Min Retransmit Time:	The minimum value permitted by a TCP implementation for the retransmission timeout, measured in milliseconds.
Max TCP Connections:	The limit on the total number of TCP connections the CSS can support.
Passive Opens:	The number of times TCP connections have made a direct transition to the SYN-RCVD state from the LISTEN state.
Resets:	The number of times TCP connections have made a direct transition to the CLOSED state from either the ESTABLISHED state or the CLOSE-WAIT state.
Input Segments:	The total number of segments received, including those received in error. This count includes segments received on currently established connections.
Retransmit Segments:	The total number of segments retransmitted--that is, the number of TCP segments transmitted containing one or more previously transmitted octets.
Output Resets:	The number of TCP segments sent containing the RST flag.
ICMP Statistics:	
Echo Requests In:	The number of received ICMP Echo (request) messages.
VIP Echo Requests:	The sending Echo request count for the VIP.
Unreachable:	The number of received ICMP Destination Unreachable messages.
Redirect:	The number of received ICMP Redirect messages.

Table 3-9 Field Descriptions for the show ip statistics Command (continued)

Field	Description
Router Solicit:	The number of received ICMP router solicitation packets.
Param Problem:	The number of received ICMP Parameter Problem messages.
Timestamp Reply:	The number of sent ICMP Timestamp Reply messages.
Information Reply:	The number of received ICMP reply packets.
Mask Reply:	The number of received ICMP Address Mask Reply messages.
Echo Replies In:	The number of received ICMP Echo reply messages.
VIP Echo Replies:	The sending Echo replies in response to echoes for the VIP.
Source Quench:	The number of received ICMP Source Quench messages.
Router Adv:	The number of received ICMP router advertisement packets.
Time Exceeded:	The number of received ICMP Time Exceeded messages.
Timestamp:	The number of sent ICMP Timestamp (request) messages.
Information Request:	The number of received ICMP information request packets.
Mask Request:	The number of sent ICMP Address Mask Request messages.
Invalid:	The number of received bad ICMP type packets.
ARP Statistics:	
Requests In:	The number of received ARP request packets.
Requests Out:	The sending ARP request packet count.

Table 3-9 Field Descriptions for the `show ip statistics` Command (continued)

Field	Description
Duplicate Addr:	The number of received ARP packets with duplicate IP address detected count. This can be the local IP address, VIP, or virtual interface
Invalid:	The number of invalid or bad ARP packets.
Replies In:	The number of received ARP reply packets.
Replies Out:	The sending ARP reply packet count.
In Off Subnet:	The number of received ARP packets with sender or target addresses outside of the subnet range of the receiving interface.
Unresolved:	The number of processed IP frames with unresolved next hop MAC addresses.

Showing IP Summary

Use the **show ip summary** command to display a summary of IP global statistics. The statistics include data on reachable and total routes, reachable and total hosts, memory in use for each, and total IP routing memory in use.

Table 3-10 describes the fields in the **show ip summary** output.

Table 3-10 Field Descriptions for the `show ip summary` Command

Field	Description
Reachable Routes	The current number of reachable routes.
Total Routes	The current number of routes maintained, both reachable and unreachable.
Reachable Hosts	The current number of reachable host entries.
Total Hosts	The current number of host entries, both reachable and unreachable.
Total Memory in use - IP Routing Memory Pool	The total amount of memory in bytes allocated for the IP routing table. When there are no additional free entries in the memory pool, more memory is allocated to the pool.

Configuring Bridging for the CSS

You can configure the following **bridge** command options for the CSS:

- **bridge aging-time** - Set the bridge filtering database aging time
- **bridge forward-time** - Set the bridge forward delay time
- **bridge hello-time** - Set the bridge hello time interval
- **bridge max-age** - Set the bridge spanning-tree maximum age
- **bridge priority** - Set the bridge spanning-tree priority
- **bridge spanning-tree** - Enable or disable the bridge spanning-tree

Configuring Bridge Aging-Time

To set the bridge filtering database aging time for the CSS, use the **bridge aging-time** command. The aging time is the timeout period in seconds for aging out dynamically learned forwarding information. Enter an integer from 10 to 1000000. The default is 300.

For example, to set the bridge aging time to 600, enter:

```
(config)# bridge aging-time 600
```

To restore the default aging time of 300, enter:

```
(config)# no bridge aging-time
```

Configuring Bridge Forward-Time

To set the bridge forward delay time, use the **bridge forward-time** command. The forward time is the delay time in seconds that all bridges use for forward delay when this bridge is acting as the root. Enter an integer from 4 to 30. The default is 4.



Note

Make sure that bridge maximum age is less than or equal to 2 x (bridge forward-time - 1 second) and greater than or equal to 2 x (bridge hello-time + 1 second).

For example, to set the bridge forward time to 9, enter:

```
(config)# bridge forward-time 9
```

To restore the default delay time of 4, enter:

```
(config)# no bridge forward-time
```

Configuring Bridge Hello-Time

To set the bridge hello time interval, use the **bridge hello-time** command. The hello time is the time in seconds that all bridges use when this bridge is acting as the root. Enter an integer from 1 to 10. The default is 1.

For example, to set the bridge hello time to 9, enter:

```
(config)# bridge hello-time 9
```

To restore the default hello time interval of 1, enter:

```
(config)# no bridge hello-time
```

Configuring Bridge Max-Age

To set the bridge spanning-tree maximum age, use the **bridge max-age** command. The maximum age is the time in seconds that all bridges use when this bridge is acting as the root. Enter an integer from 6 to 40. The default is 6.



Note

Make sure that bridge maximum age is greater than or equal to 2 x (bridge hello-time + 1 second) and less than or equal to 2 x (bridge forward-time - 1 second).

For example, to set the bridge maximum age to 21, enter:

```
(config)# bridge max-age 21
```

To restore the default maximum age of 6, enter:

```
(config)# no bridge max-age
```

Configuring Bridge Priority for the CSS

To set the priority that spanning tree uses to choose the root bridge in the network, use the global **bridge priority** command. In spanning tree, the 2-octet field is prepended to the 6-octet MAC address to form an 8-octet bridge identifier. The device with the lowest bridge identifier is considered the highest priority bridge and becomes the root bridge. The range for bridge priority is 0 to 65535. The default is 32768.

For example:

```
(config)# bridge priority 1700
```

To restore the bridge priority to its default of 32768, enter:

```
(config)# no bridge priority
```

Enabling and Disabling Bridge Spanning-Tree

Bridge spanning-tree is enabled by default. To disable spanning-tree, enter:

```
(config)# bridge spanning-tree disable
```



Caution

Disabling spanning-tree may make your network susceptible to packet storms.



Note

When spanning-tree is disabled, the CSS drops Bridge Protocol Data Units (BPDUs).

To reenable bridge spanning-tree, enter:

```
(config)# bridge spanning-tree enable
```

Showing Bridge Configurations

The CSS enables you to show the bridge forwarding and bridge status information.

To display bridge forwarding information, use the **show bridge forwarding** command. Table 3-11 describes the fields in the **show bridge forwarding** output.

Table 3-11 *Field Descriptions for the show bridge forwarding Command*

Field	Description
VLAN	The bridge interface virtual LAN number
MAC Address	The MAC address for the entries
Port Number	The port number for the forwarding

To display bridge status information, use the **show bridge status** command. Table 3-12 describes the fields in the **show bridge status** output.

Table 3-12 *Field Descriptions for the show bridge status Command*

Field	Description
STP State	The state of the spanning-tree protocol, enabled or disabled.
Root Max Age	The timeout period in seconds of the host for timing out root information.
Root Hello Time	The interval in seconds that the root broadcasts its hello message to other devices.
Root Fwd Delay	The delay time in seconds that the root uses for forward delay.
Designated Root	The bridge ID for the designated root.
Bridge ID	The bridge ID of this bridge.
Port	The port ID.

Table 3-12 Field Descriptions for the `show bridge status` Command (continued)

Field	Description
State	<p>The state of the port. The possible states are:</p> <ul style="list-style-type: none"> • Block, the blocking state. A port enters the blocking state after switch initialization. The port does not participate in frame forwarding. • Listen, the listening state. This state is the first transitional state a port enters after the blocking state. The port enters this state when STP determines that the port should participate in frame forwarding. • Learn, the learning state. The port enters the learning state from the listening state. The port in the learning state prepares to participate in frame forwarding. • Forward, the forwarding state. The port enters the forwarding state from the learning state. A port in the forwarding state forwards frames. • Disabled, the disabled state. A port in the disabled state does not participate in frame forwarding or STP. A port in the disabled state is non operational.
Designated Bridge	The bridge ID for the designated bridge.
Designated Root	The bridge ID for the designated root.
Root Cost	The cost of the root.
Port Cost	The cost of the port.
Desg Port	Designated port.

Configuring Secure Shell Daemon

Secure Shell Daemon (SSHD) is a server program designed to log into another computer over a network, execute commands in a remote machine, and move files from one machine to another machine. It provides strong authentication and secure communications over non-secure channels. SSHD is intended as a replacement for rlogin, rsh, and rcp.

**Note**

This feature requires an SSHD Server License, which enables SSHD functionality on both the Standard and Enhanced versions of CSS software.

SSHD protects against:

- Attacks from machines pretending to be another server, router, or a domain name server
- IP spoofing, where a remote host sends out packets that pretend to come from another trusted host
- IP source routing, where a host can pretend that an IP packet comes from another trusted host
- DNS spoofing, where an attacker forges name server records
- Interception of clear text passwords or data on the network
- Manipulation of data by people in control of intermediate hosts

**Note**

To enhance security when using SSHD, disable Telnet access. To disable Telnet access, use the **telnet-access disable** command as described later in this chapter. Telnet access is enabled by default.

The CSS provides the following SSHD commands:

- **sshd keepalive** - Enable SSHD keepalive
- **sshd port** - Set the SSHD port
- **sshd server-keybits** - Set the number of bits in the server key

For more information on these options and associated variables, refer to the following sections:

- Configuring SSHD Keepalive
- Configuring SSHD Port
- Configuring SSHD Server-Keybits
- Disabling and Enabling Telnet Access when using SSHD

Configuring SSHD Keepalive

To enable SSHD keepalive, use the **sshd keepalive** command. SSHD keepalive is enabled by default.

For example, to enable SSHD keepalive:

```
(config)# sshd keepalive
```

To disable the SSHD keepalive, enter:

```
(config)# no sshd keepalive
```

Configuring SSHD Port

To set the port number to which the server listens for connections from clients, use the **sshd port** command. Enter a port number from 22 to 65535. The default is 22.

For example, to configure port number 57:

```
(config)# sshd port 57
```

To reset the port number to the default of 22, enter:

```
(config)# no sshd port
```

Configuring SSHD Server-Keybits

To set the number of bits in the server key, use the **sshd server keybits** command. Enter the number of bits from 512 to 65535. The default is 768.

For example, to set the number of bits to 919:

```
(config)# sshd server-keybits 919
```

To reset the number of bits to the default of 768, enter:

```
(config)# no sshd server-keybits
```

Disabling and Enabling Telnet Access when using SSHD

When you use SSHD, you may wish to disable non-secure Telnet access to the CSS. Use the global **restrict telnet** command to disable Telnet access to the CSS. Telnet access is enabled by default.

For example, to disable Telnet access, enter:

```
(config)# restrict telnet
```

To reenable Telnet access to the CSS, enter:

```
(config)# no restrict telnet
```

Showing SSHD Configurations

To display SSHD configurations, use the **show sshd config** command. Table 3-13 describes the fields in the **show sshd config** output.

Table 3-13 Field Descriptions for the **show sshd config** Command

Field	Description
Keepalive Setting	Whether or not SSHD keepalive is enabled. SSHD keepalive is enabled by default.
No. of Server Key Bits	The number of bits in the server key. The default is 768. The range is from 512 to 65535.
Listen Port No.	The port number that the server listens to connections from clients. The default is 22. The range is from 22 to 65535.
Telnet Disallowed	Whether or not Telnet access to the CSS is allowed. Telnet access is enabled by default.

Configuring Opportunistic Layer 3 Forwarding

The CSS opportunistic Layer 3 forwarding feature allows the CSS to reduce the number of network device hops for certain packets or flows. The CSS forwards packets at Layer 3 if the destination MAC address in the Ethernet header is the CSS's MAC address. Opportunistic Layer 3 forwarding allows the CSS to make Layer 3 forwarding decisions even if the layer 2 packet destination MAC address does not belong to the CSS.

For example, Figure 3-1 shows a CSS connected to VLAN1 and VLAN2. Each VLAN has an end station and an uplink to Router1. End stations A and B both point to Router1 as their default router. When end station A transmits a packet to end station B, it uses its default route to Router1. The packet contains Router1's destination MAC address. A traditional layer 2 device would forward the packet to Router1 and it would forward the packet to end station B on VLAN2.

Opportunistic Layer 3 forwarding provides three modes of operation:

- **local (default)** - Apply opportunistic Layer 3 forwarding if the destination IP address belongs to a node that resides on one of the subnets directly attached to the CSS *and* the CSS knows an ARP resolution for that node. Because the local option is the default, use the **no ip opportunistic** command to reconfigure **ip opportunistic** to local.
- **all** - Apply opportunistic Layer 3 forwarding if the destination IP address matches any routing entry on the CSS. This mode is not recommended if the topology includes multiple routers and the CSS does not know all of the routes that the routers know.
- **disabled** - The CSS does not perform opportunistic Layer 3 forwarding. Regular Layer 3 forwarding is performed only for packets that contain the CSS's destination MAC address.

For example, to configure ip opportunistic Layer 3 forwarding to **all**, enter:

```
(config)# ip opportunistic all
```

To reconfigure ip opportunistic Layer 3 forwarding to the default of **local** enter:

```
(config)# no ip opportunistic
```

When you configure **ip opportunistic all**, you can use the **ip route originated-packets** command to configure routes that the CSS will use to reach devices, but will not use as opportunistic routes for forwarding traffic. Routes created using the **ip route originated-packets** command apply only to packets that originate on the CSS. Packets and flows forwarded by the CSS will not use these routes.

For example,

```
(config)# ip route 0.0.0.0/0 192.168.1.7 originated-packets
```

Where to Go Next

For information on configuring circuits and interfaces, refer to Chapter 4, Configuring Interfaces and Circuits.

